An Innovative Web-based System for high Lossless Compression and Fast, Interactive Transmission of Visible Human Color Images

Gilberto Zamora[1] , Mark Wilson[1], Sunanda Mitra[1], George Thoma[2]

[1]Computer Vision and Image Analysis Lab
Department of Electrical Engineering
Texas Tech University, Lubbock, TX 79409

[2]Lister Hill National Center for Biomedical Communications
National Library of Medicine

gzamorac@ttacs.ttu.edu

Abstract

This paper describes a system uniquely designed for the management and distribution of the Visible Human (VH) cryosectioned color images of the male and the female. Moreover, this design approach is applicable to the management and distribution of other large image datasets when optimized to their specific characteristics. This research work includes the design and implementation of a robust lossless compression algorithm to store and transmit VH images over the Internet, while also allowing easy access to the images through the interactive selection of any section of the database for downloading and viewing in the users' machines, accomplished by an interactive Java applet. Our system design is based on a color transformation from RGB to HSI to facilitate an automated extraction of an anatomic object from the background yielding a segmented image prior to compression. This unique two-step segmentation process, followed by lossles coding using an Adaptive Arithmetic Coding (AAC) model, yields an average data reduction of 7.8:1 for both male and female datasets and as high as 16:1 for specific slices. Reconstruction of the compressed images is achieved by applying the inverse AAC algorithm. The final step is to set the background to black and display, or store, the reconstructed images.

The system described here presents a solution for storage, management and distribution of a unique large image dataset, namely VH color images, by efficient two step lossless compression, content based retrieval and interactive fast transmission and viewing. Such a system as described here provides innovative concepts for designing systems for other large databases by using optimization criteria for specific databases.

Keywords: Visible Human, lossless compression, color transformation, content-based segmentation, web-based software.

Introduction

Large volumes of image or other databases demand efficiently designed systems for management and use. Such designs involve image/data analysis and communications methodologies including data classification, segmentation, compression, fast transmission of needed data files, and content-based retrieval of information for viewing by the users.

For the VH color datasets theoretical computations yield transmission times of about 3.3 days over a traffic-free T1 line for the 15GB Male, 40GB Female VH low resolution (2048 x 1216 pixels) color images [2]. Since traffic-free conditions are virtually impossible, the actual transmission time over a T1 line is considerably longer than the time computed under ideal conditions. This is the reason why reduction in file size is required to allow users to have access to and store these datasets in a practical and efficient way. In addition to having reduced size datasets, it is important to let users around the world have access to the VH images and the ideal way to do it is through the Internet. Therefore, one of the objectives is the design and implementation of Web-based software that can be used to select and download the files from the datasets that users are interested in. This application specific, platform independent, software is constructed by a graphical user interface between the user and the web server, that stores the files, and allows the selection, preview and downloading of the selected compressed files from the NLM web server to the user's computer. The viewer software also allows the user to decompress and view the previously downloaded files.

Although significant reduction in file size is one of the objectives, this image compression process should not involve any loss of anatomical information, for archiving purposes, or the loss of any of the fiduciary marks used to register the VH images. On the other hand, traditional lossless compression algorithms [3] yield compression ratios (CR) of up to 3:1 only thus representing no significant improvement in the problem at hand. Therefore, a custom designed lossless compression algorithm was developed to reach higher CRs than traditional methods while taking into account the special characteristics of the VH color images [4]. This method performs a segmentation of the image based on its content i.e. background, anatomy, fiduciary marks, and passes this information to an Adaptive Arithmetic Coder (AAC) [6] that performs the lossless compression. Initial experimental results yielded an average CR of 9.2:1 for the male VH color image data used.

The significance of this project is directly related to the number of people who would eventually be able to use it. The users around the world access the Internet with a number of different platforms. The Java programming language is designed precisely to overcome this problem because it allows the programmer to design and deploy applications using code that is directly transportable along the platforms supported by the language. The software described in this report was written in Java 1.2 and is specifically designed for Windows and Unix based systems but can be easily transported to other Java-enabled systems.

Current access to the VH image datasets is done by file name but it is difficult for most of the users to relate these names to the actual region of the body the files correspond to. Therefore one of the goals was to design an interactive Graphic User Interface (GUI) such that the user could choose the files from a "body map", i.e. a full body color image, using a simple selection window. Other desirable graphic characteristics, like preview capability and file name display, were also included in the design of this GUI.

As an important part of the project, the main characteristics, advantages, and disadvantages of two lossy coding algorithms, JPEG and EZW were also investigated [8][9]. This study shows the differences in quality, compression ratio, and decoding time between these two algorithms. These three variables are used to determine which lossy algorithm would be better to use as an option if the user would prefer to download lossily compressed images. Although this would mean a relative saving in time, would also imply a certain loss of detailed visual information. Therefore, the focus of this project remained in the use of lossless compression. A vector quantization-based high fidelity lossy compression algorithm, namely AFLC-VQ, was not included in the present study because of the longer computational time involved [10][11].

Methods

Lossless Compression

The methodology used in this research work is to apply a contour detection scheme to segment the image as the first step. An adaptive arithmetic model [2] is then applied to code the segmented image losslessly. For the male Visible Human color image set, the overall average lossless compression using the above scheme is around 9:1 whereas the compression ratio of selected slices can be as high as 16:1. The achievable compression ratio depends on the actual bit rate of the segmented images attained by lossless coding as well as the compression obtained from segmentation alone. This significant improvement in lossless compression achievable by segmentation and entropy coding has the advantage that the computational time required by the entire process is quite fast. Therefore this improved lossless coding model can be easily applied to large medical images such as the VH color images.

Medical images present a great source of development for segmentation techniques due to the fact that, generally, many different regions within the images can be distinguished, classified, and labeled depending on specific applications. However, feature segmentation, can also be used to determine sections of the images useful for subsequent processing steps from sections that are not. This point of view leads to the idea of determining regions of the images where there is no relevant information for the next processing steps, i.e. it is redundant, and this is the basic idea behind data compression. Then, the problem becomes the implementation of a technique to distinguish those regions of the image that can be set aside in the processing flow.

It has been shown that a color image can be represented in terms of several color spaces and that these representations are helpful in performing a number of different tasks like displaying, broadcasting, compression, and transmission [2][4]. In this respect, the Hue component, from the Hue-Saturation-Intensity (HIS) color space [7], offers the advantage of representing colors as angles in the unitary circle. This can be used to distinguish different parts of images, for example, backgrounds from the main regions of interest. Visible Human color images [1][5] are a good example of the application of such a technique because the background can be distinguished very easily using its Hue component. In these images, the background carries no information and therefore it can be discarded for further image processing. By doing this, some compression is reached already. One step further is to code the remaining information in a compact way using entropy coding techniques to obtain data representations suitable for storage and/or transmission purposes without any loss of the original data. Among lossless coding techniques, arithmetic coding has been proven to have performance advantages, and with the inclusion of adaptive models and predictive coding, compression ratios (CR) very close to the limits imposed by source entropy can be obtained [2].

Image segmentation

For the case of the Visible Human color images, the process consists of finding a segmentation mask to discard the background. The Hue component of each image is used as the basis of this mask. The Hue value of each RBG pixel can be obtained from the following relationship [4]

$$H=\cos^{-1}\left[\frac{0.5\cdot((R-G)+(R-B))}{\left[(R-G)^{2}+(R-B)\,(G-B)\right]^{\frac{1}{2}}}\right]$$

$H=360^{o-}H$, if $B>G$

where R, G, and B are the red, green, and blue values of the corresponding pixel. The following three figures show how the hue component of the VH color images can be used to discard the background. Figure 1 shows the original RGB image (image a_vm1126.raw) and a horizontal line at row 590. Figure 2 shows the original pixel values along this line, i.e. the profile of the image at row 590. Then, figure 3 shows the hue component of this profile. It can be clearly seen that the hue component corresponding to the anatomical parts is completely different to the hue component corresponding to the surrounding background and that a simple threshold would separate these two areas. This suggests the construction of a segmentation mask.

This segmentation mask is built in two steps. First, the Hue values are adjusted in contrast and brightness to accentuate the difference between the anatomical parts and the rest of the image. The next step is to apply a threshold to the hue image. This operation yields a binary image because the original dynamic range is regrouped into two values, black (0) for the values between zero and the threshold, and white (255) for the values above the threshold. It can be seen from figure 3 that this binarization produces black areas corresponding to the anatomical parts and white areas for the rest of the image. Because the objective is to mask the anatomical parts the negative of the binary image is taken to have white areas corresponding to anatomical parts and black areas corresponding to the rest of the image. This binary image is the segmentation mask that is applied to the original image by performing a logical AND operation in a pixel by pixel basis. The result is that all the original pixels that correspond to black pixels in the mask become black and those corresponding to white pixels in the mask keep their original value. Figure 4 shows the segmentation process step by step.

In the case of Visible Human images, the segmented images still have zones that remain after the previous process that can be discarded, like the gray scale and post-it note at the bottom of the image and the boundaries of the ice containing the anatomic parts. Therefore additional segmentation windows can be used as the final step in the process. These windows are rectangular boundaries that surround the parts of interest in the image, i.e. anatomy and fiduciary marks, as shown in figure 5. The result of the segmentation process is a new image without background and reduced in size because anything outside the segmentation windows is discarded.

The segmented image can then be subjected to subsequent processing steps, such as lossless coding. Even direct storage results in substantial compression ratios. For example, for the image in figure 5, the compression ratio after the segmentation step is 4.63:1 already.

Adaptive Arithmetic coding

The lossless coding technique chosen to compress the segmented images is Adaptive Arithmetic Coding because of its already proven advantages over other techniques like Huffman, Lempel-Ziv, and lossless JPEG [3][7]. The principle of this technique is to represent data from the source as a real number, within an initial range, and having this number related to the probability of occurrence of the sequence of symbols to be coded. The differentiation between source data and coded symbols is important because they might not be the same. In the early implementations of arithmetic coding, the data from the source was ASCII characters, pixel values, or any other representation of the source, and those values were taken as the symbols to be coded. However, the implementation used in this project included coding of the difference (symbol) between adjacent pixels (data). This modification is useful in cases of low frequency signals, like images, because it yields a reduction of the range of the symbols representing the image. A reduction in this range yields, in turn, an increase in compression ratio. Another characteristic of the actual implementation of the Arithmetic coding process is to have an adaptive model of the source. An adaptive model means that the model of the source, i.e. the image, is updated with each new symbol that is coded. This allows the use of the algorithm with any image in the dataset [2]. Although the compression ratios obtained with this technique depend on each image, an average CR of about 2:1 [2] was observed when it was applied to Visible Human images before segmentation. As an example, when the Adaptive Arithmetic Coding was applied to the segmented image in figure 5, the CR was 8.56:1.

Statistical Error Analysis

Lossless Coding

The traditional statistical quality measures such as the mean square error (MSE) and the peak signal to Noise ratio (PSNR) are not applicable to lossless coding since by definition the encoding process does not introduce any error. Therefore the decoded image should be the same as the original. The missclassification rate of the pixels in the segmented image can not be computed since no truth models of the anatomical parts of the Visible Human color images are currently available. However, the elimination of the blue regions within the anatomical parts is justified since these blue regions are considered to be artifacts caused by the blue gel interjected into the Visible Human bodies during the preparation prior to slicing.

Lossy Coding

In order to achieve large compression ratios for fast transmission of images some information must be sacrificed. An ideal algorithm makes better use of the losses so that they are less apparent to the user. In the following a comparison is be made between the widely used standard JPEG (Joint Photographic Experts Group) [8] and a newer technique EZW (Embedded Zerotrees Wavelet [9]. The comparison will include a statistical analysis and a list of advantages and disadvantages the algorithm possesses.

The problem at hand is to obtain the best image quality for a given bit rate in the shortest amount of time. Currently image quality is measured quantitatively with statistics such as PSNR (Peak Signal to Noise Ratio) and MSE (Mean Squared Error) or qualitatively by a human expert. These statistics measure how the compressed image differs from the original through a pixel by pixel comparison. The distortions created in the images are called artifacts. The main artifact noticed qualitatively in JPEG is the blocking artifact. This occurs since the algorithm operates on 8x8 blocks by applying the DCT (Discrete Cosine Transform) transform then using scalar quantization on these individual blocks [8]. The EZW algorithm does not posses this artifact because it does not break the image down into small blocks, instead a wavelet transform is applied to the entire image creating a multi-resolution data pattern and a scalar quantization technique is used to compress [9]. The main artifact in EZW is a blurring effect, which is more soothing to the eyes than sharp blocks. Examples of both artifacts are pictured in figure 6. Another disadvantage of the JPEG algorithm is its variable code rate in which the level of compression differs from image to image. With the EZW algorithm the code rate can be assigned a priori. Therefore the entire Visible Human color dataset can be compressed at the same predetermined assigned rates. When comparing the run times of EZW and JPEG the JPEG algorithm performs much faster. This is because the problem is divided into 8x8 blocks which can easily fit into any processor's cache memory, while the EZW algorithm must utilize RAM in order to operate on the entire image. The EZW algorithm also requires more computations thus slowing it down. As RAM and processors increase in speed this time gap is narrowing quickly. In order to compare the two algorithms on the Visible Human Dataset, 6 female and 6 male images were selected. Two comparisons were made, one with the background segmented leaving the fiduciary marks, and one with the background included. The results show that quality improves for both algorithms when the background is segmented. The statistical analysis quantitatively proves that EZW gives better image quality than JPEG at high compression ratios, while they perform similarly at low compression ratios. The decompression time for JPEG is much faster than that of EZW.

The experimental compression ratios were chosen by first compressing the images with JPEG at two different quality factors. The compression ratio was then calculated dividing the original file size by the compressed file size. These compression ratios were used to compress the same images using EZW. figure 7 shows the results of Mean Square Error (MSE) calculation for this experiment and figure 8 the compression ratios obtained.

Based on these results the JPEG and EZW algorithms perform statistically the same. The main advantage of the JPEG algorithm is its decompression speed that is much quicker than EZW. The artifacts for both algorithms are not visible in the high quality images and barely visible in the low quality. Our statistical analysis shows no difference in performance between JPEG and EZW. A qualitative visual analysis shows that the artifacts are barely present for the compression ratios chosen concluding neither algorithm has an advantage. Therefore, the main advantage here is the decompression time where JPEG has a clear lead. Therefore JPEG is proposed to lossily compress the segmented visible human male and female detasets.

Overall System Performance Analysis for Fast Transmission

An overall performance analysis of the system was done using the compressed test images. This analysis involved the actual decoding time for lossily and losslessly compressed images as well as the calculated transmission time over a traffic-free T1 line. Figure 9 shows the performance analysis from which it can be seen that the time difference between the use of lossily and losslessly compressed images is not significant compared to the advantage of having perfect reconstruction from the losslessly compressed VH images.

Java software design

Java was selected as the programming language for viewing and display in order for the program to be usable by a large public. A program was needed for the user to correlate the image selection to a section of the body. To do this, a picture of the whole body (male or female) is displayed with a superimposed selection window that can be positioned and resized using the user's mouse. The user can also see preview images from the top and bottom ranges of the selection box. These preview images are highly JPEG-compressed images ($CR \approx 200:1$) that are resized and displayed any time the user selects the "Update Preview" button. These preview images may be viewed at full resolution by selection of the "Top Preview View" or "Bottom Preview View" buttons. In order to fine tune the selection a "Up 2 Slices" or "Down 2

Slice” button can be used. In addition, the text file names of the selected images are also displayed. In the menu bar the user then has the ability to choose the quality level of the image. There are two choices of compression, the perfectly lossless quality or the degraded lossy quality with smaller file size. These are the files that are downloaded by the user once they select to download from the pull down menu. Once they choose to download, the native “file save” dialog box appears for the user to select the directory in which to save the image files. The menu bar also has the menu to select between the male or female data sets. Figure 10 shows a snapshot of the selection/downloader Java applet with the selection tools highlighted.

The process of selection and downloading is done on-line but the viewing part is performed off-line. That is, a user has to be connected to the Internet to select the images of one’s interest and download them into the user’s computer, but once they are stored the images can be decompressed and viewed off-line. That is why there is another application, called the “viewer”, that allows the user to read in compressed images (jpeg or aac), decompress, and display them. Then the decompressed images can be stored again in “raw” format if desired so these can be used by other commercial imaging packages. Figure 11 shows a snapshot of the viewer.

Block diagram of the system

Figure 12 shows the block diagram of the system implemented for this project. The web server at NLM stores the compressed files along with the code for the selection/downloader applet and the viewer application. The user, through the Internet, can download the code to run the applet on-line and the viewer off-line. He can also download and store the files of the compressed images and view them using the viewer.

Results

Actual transmission times for original and compressed VH images were recorded in order to address the advantages of the system. The following were the characteristics of the test:

- Test images:
  - Original images:
    - A_vm1091.raw, file size: 7,471,104 bytes, 7.4 MB
    - A_vm1421.raw, file size: 7,471,104 bytes, 7.4 MB
    - A_vm2417.raw, file size: 7,471,104 bytes, 7.4 MB
  - Compressed images:
    - Vm1091.aac, file size: 515,545 bytes, CR=14.49:1
    - Vm1421.aac, file size: 2,011,379 bytes, CR=3.71:1
    - Vm2417.aac, file size: 648,704 bytes, CR= 11.52:1
- Method of transmission
  - ftp from ceb.nlm.nih.gov to ee066.dhcp.ttu.edu
  - binary mode
- Time of day for transmission: 8 am, 10 am, 12 pm, 2 pm, 4 pm, 10 pm.

Figure 13 shows the images used for testing.

Is to be noted that during the peak traffic load for T1 line, i.e. from 2 p.m. to 4 p.m., the transmission time taken by the compressed image is approximately 4 to 56 times faster that the raw image as shown in figure 14. Figure 15 shows the corresponding graphs.

System installation

An important feature of the system is its easy installation. First, the user has to use his web browser to access the home page of the system. This home page contains all the required instructions to obtain and run the downloader Applet and Viewer application. The installation is a straightforward process that requires to create a new subdirectory, download three self-extracting files and run the corresponding batch file. After this, the user can run the Applet to select and download compressed images and view them using the viewer application.

Conclusions

From the results presented in the previous section it is clear that the system accomplished the goal of reducing the size of the files of the VH datasets in a significant way. Actual calculations yield an average CR of 7.83:1 for both the male and female datasets. The difference between the experimental and actual CRs lies in the fact that for experimentation just selected parts of the male dataset were used. However, the use of the female dataset has two impacts; firstly, the anatomical parts of the female occupy a larger percentage of the total image compared to the male, i.e. there is less background to discard, and secondly, the density of images per cm is three times larger for the female dataset.

The use of the interactive selection/downloader Applet allows the user to easily browse the male and female datasets and get the images that are of interest. The use of the off-line viewer allows the user to keep the compressed versions of the images or save them in a commercial format. These four characteristics of the system, namely reduced size of the images, reduced transmission times, interactivity, and off-line viewing, will attract more users from around the world and with multiple interests to access this important medical data set.

Another advantage of the system is that it can be used with the high resolution RGB VH images with minor modifications. These modifications involve

1. Scale size of images to the new size.
2. Scale coordinates and size of segmentation windows to the new coordinates and size. This step is simpler if the low resolution images have their corresponding counterparts in the high resolution datasets.
3. Change the name of the images to be processed in the adaptive arithmetic coding source codes
4. Take new statistics on compression ratios and transmission times.

References

[1] Zamora G. and Mitra S., "Lossless Coding of Color Images using Color Space Transformations", Proceedings of the 11th IEEE Symposium on Computer-Based Medical Systems, pp. 13-18, June 12-14, 1998.

[2] Zamora, G., Yang, S., Wilson, M., and Mitra S., "Segmentation by Color Space Transformations prior to Lifting and Integer Wavelet Transformation for Efficient Lossless Coding and Transmission", Proceedings of the 4th IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 136-140, April 2-4, 2000, Austin, Texas.

[3] Wells, R. B., "Applied Coding and Information Theory for Engineers", Prentice Hall, 1999.

[4] Gonzalez, R. C., and Woods, R. E., "Digital Image Processing", pp. 229-35, Addison-Wesley, USA, 1993.

[5] National Library of Medicine, "The Visible Human Project", web page: http://www.nlm.nih.gov/ research/visible/visible_human.html, USA, 1999.

[6] Witten, Ian H., Neal, Radford M., and Cleary, John G., "Arithmetic Coding for Data Compression". Communications of the ACM, V.30, No.6, pp.520-540, 1987.

[7] Meadows, S., Thoma, G., Long, R., and Mitra, S., "Entropy Encoding of Difference Images from Adjacent Visible Human Digital Color Photographic Slices for Lossless Compression", Proceedings of SPIE, Medical Imaging 1997, pp. 749-55.

[8] Pennebaker, W. B., and Mitchell, J. L., "JPEG Still Image Data Compression Standard", Van Nostrand Reinhold, USA, 1993.

[9] Shapiro, J. M., "Embedded Image Coding Using Zerotrees of Wavelet Coefficients", IEEE Transactions on Signal Processing, Vol. 41, No. 12, D93, pp. 3445-3462.

[10] Mitra, S. and Yang, Shu-Yu, "High Fidelity Adaptive Vector Quantization at Very Low Bit Rates for Progressive Transmission of Radiographic Images", Journal of Electronic Imaging, Vol.8, No. 1, pp 23-35, 1999.

[11] Mitra, S. , Yang, S., Zamora, G., Wilson, M., and Thoma, G., "Wavelet-Based Adaptive Vector Quantization for High-Fidelity Compression and Fast Transmission of Visible Human Color Images", paper presented at the Second Visible Human Project Conference, NIH, Bethesda, Maryland, October 1-2, 1998.

Figures

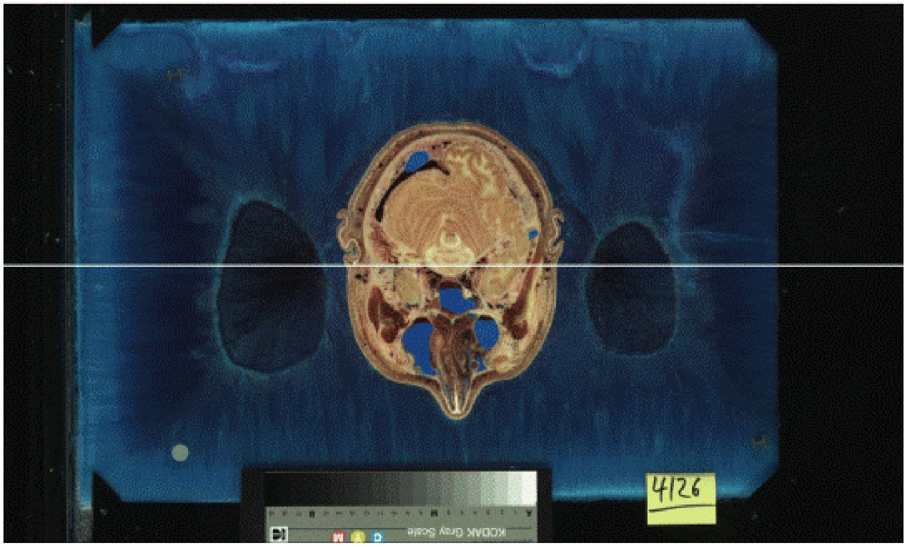


Figure 1. Original VH image and horizontal profile at row 590.

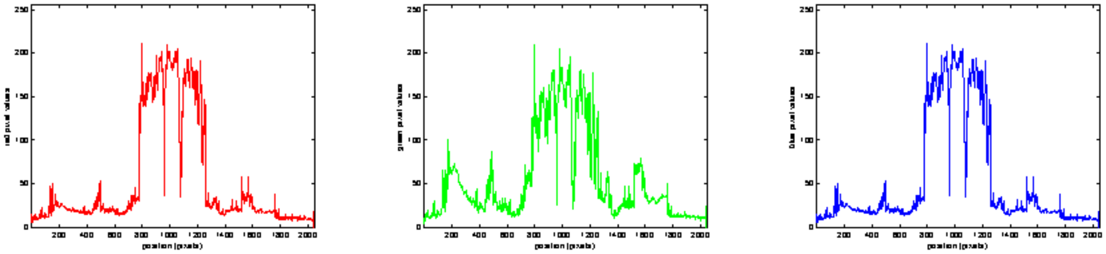


Figure 2. Red, green, and blue pixel values along the image profile (row 590).

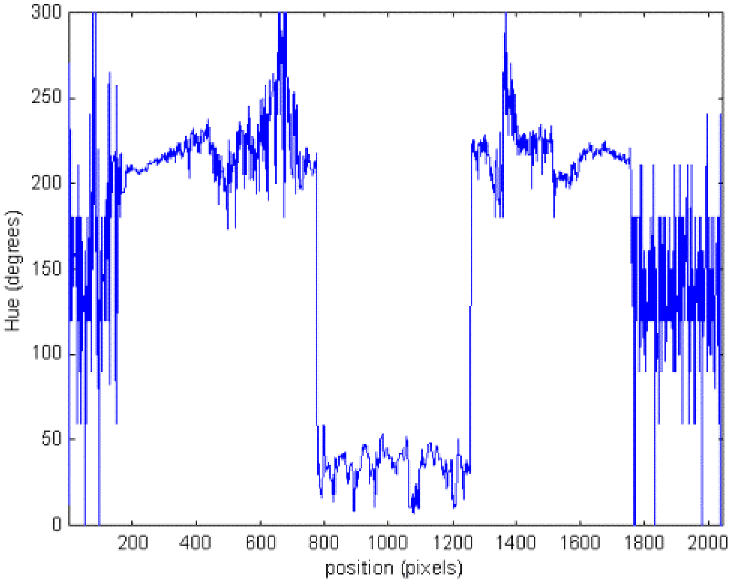


Figure 3. Hue values along the image profile (row 590).

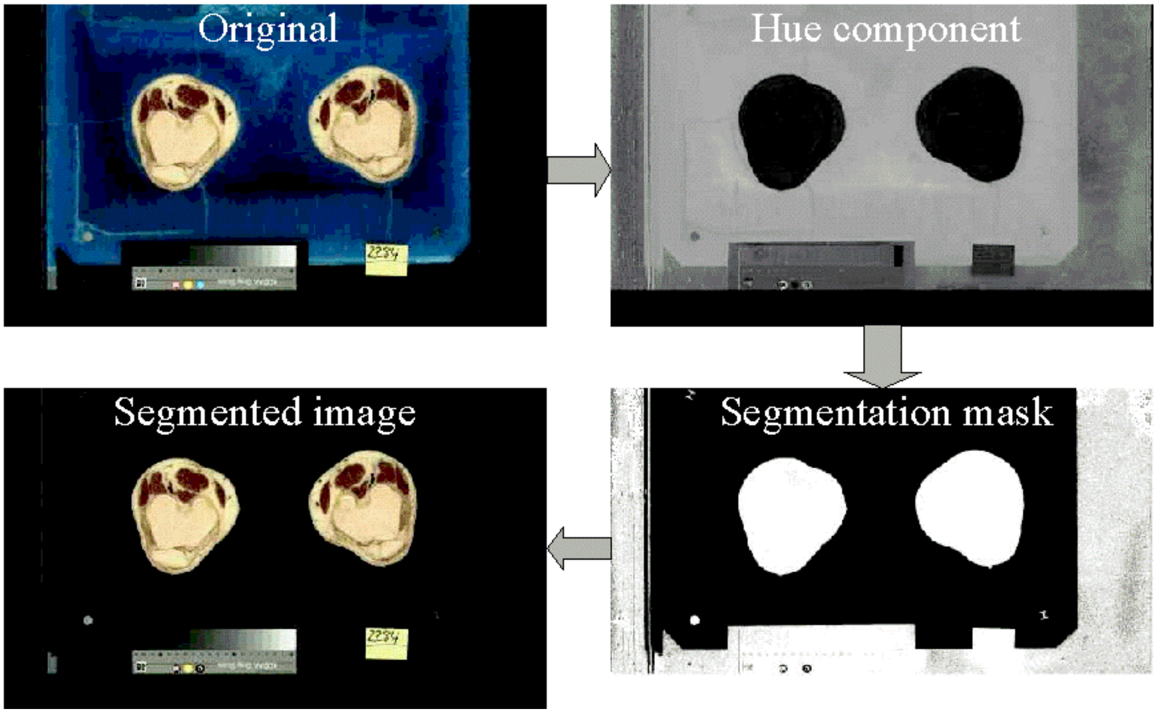


Figure 4. Graphical steps of segmentation.

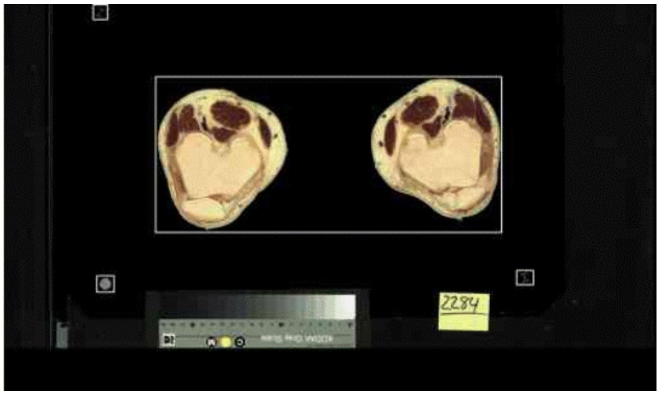


Figure 5. Example of segmentation windows.

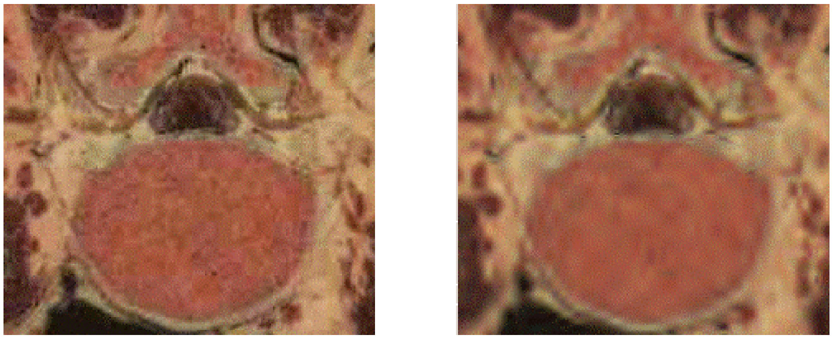
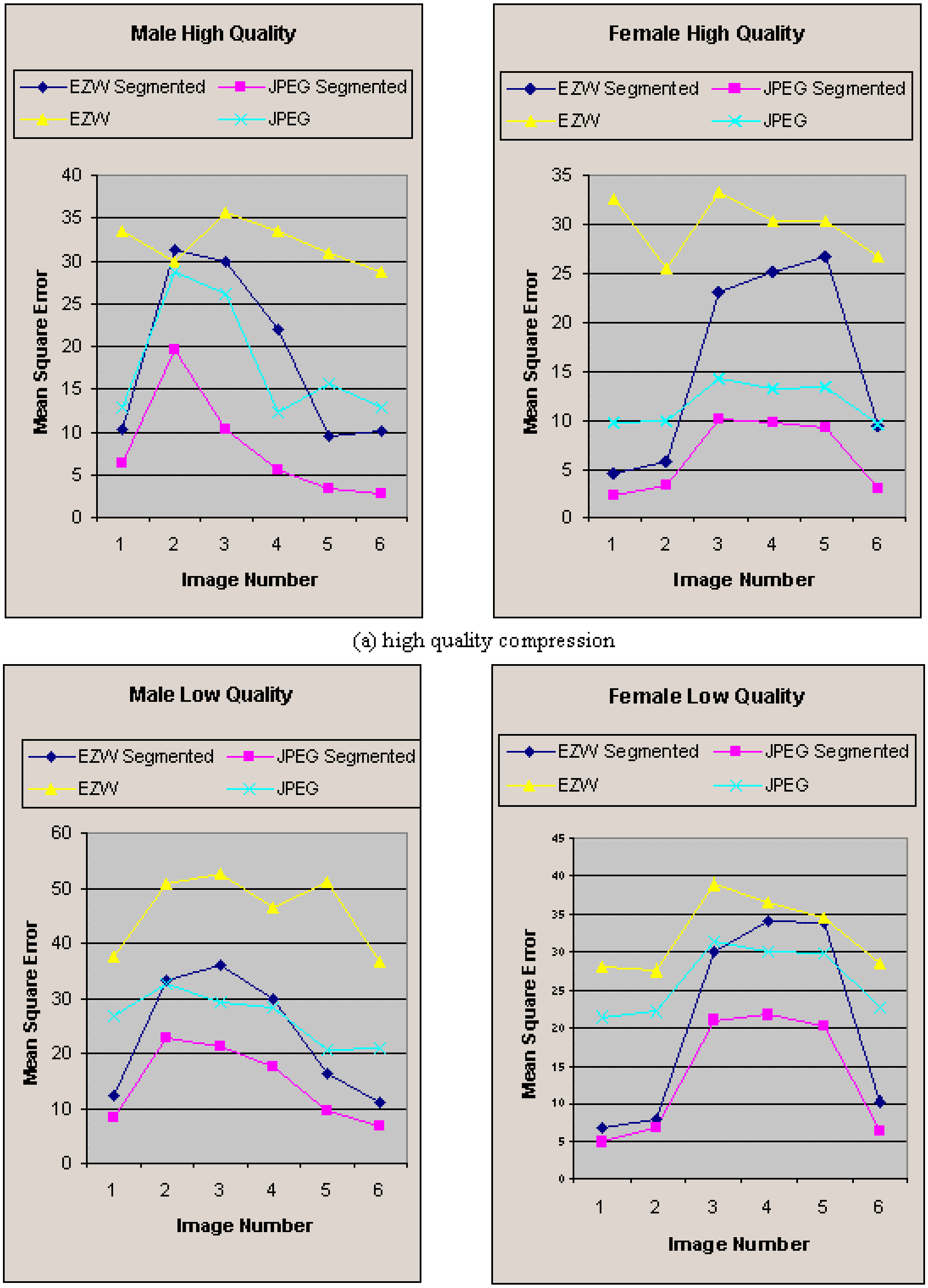


Figure 6. Crop of low quality (CR=140:1) showing the JPEG blocking (left) and EZW smoothing (right) artifacts.



(a) high quality compression

(b) Low quality compression

Figure 7. Statistical analysis of male and females images at two different CRs.

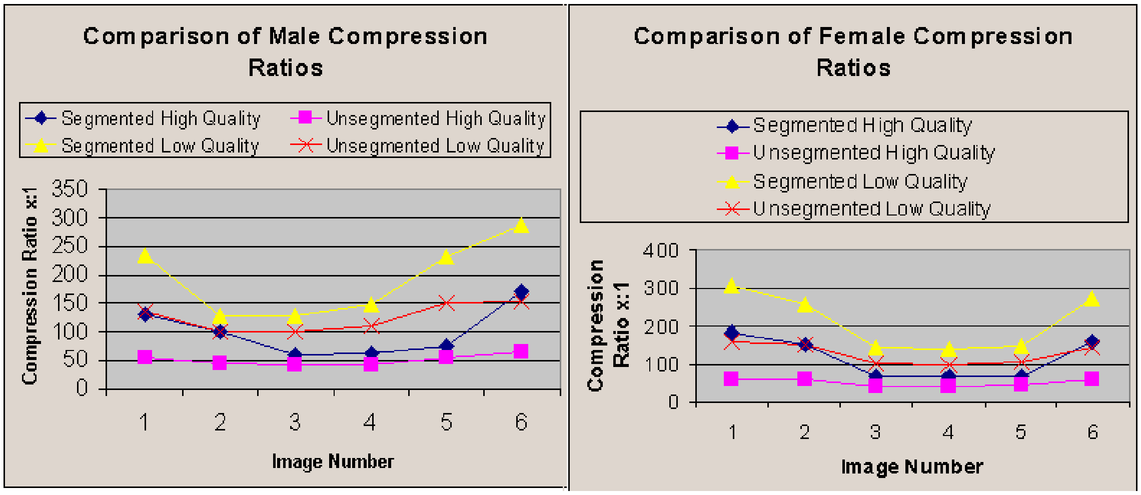
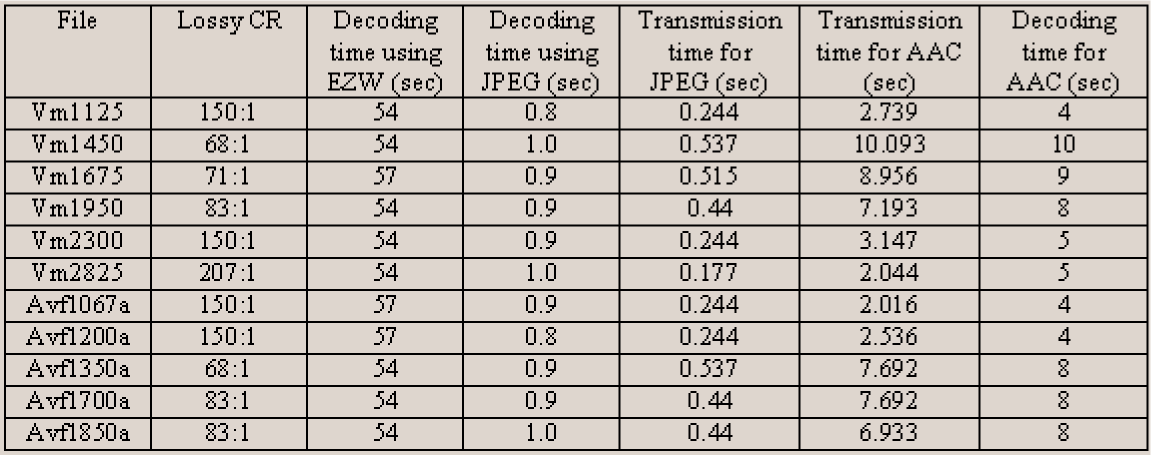
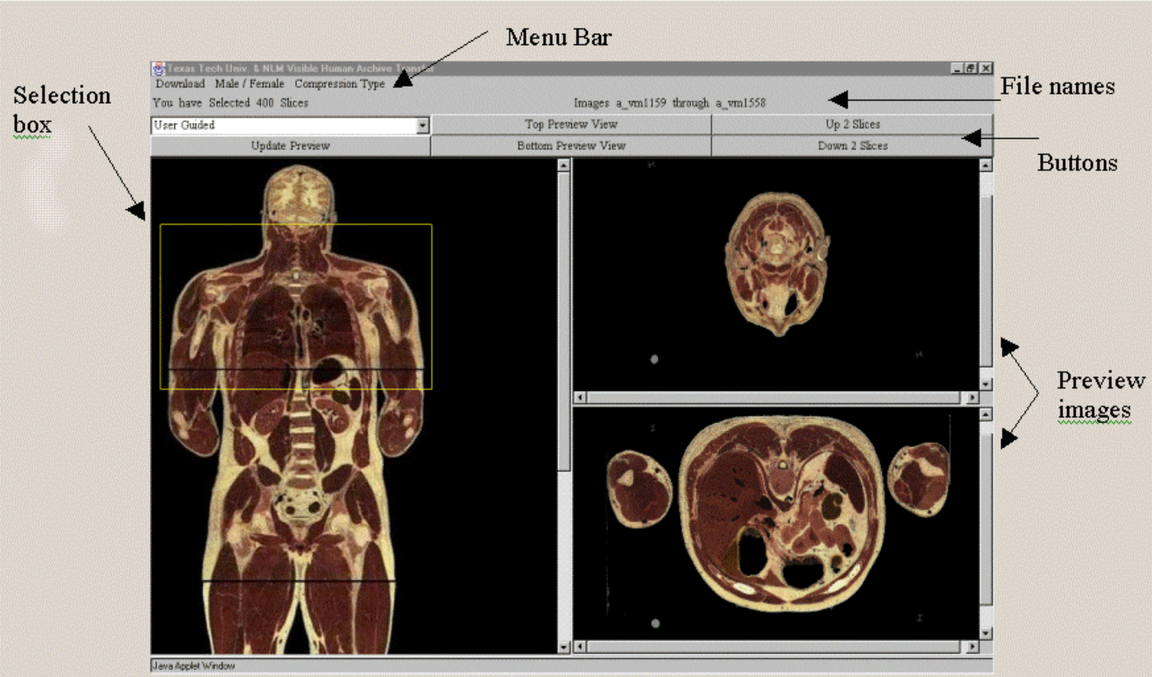


Figure 8. Comparison of compression ratios for male and female images using EZW.



File	Lossy CR	Decoding time using EZW (sec)	Decoding time using JPEG (sec)	Transmission time for JPEG (sec)	Transmission time for AAC (sec)	Decoding time for AAC (sec)
Vm1125	150:1	54	0.8	0.244	2.739	4
Vm1450	68:1	54	1.0	0.537	10.093	10
Vm1675	71:1	57	0.9	0.515	8.956	9
Vm1950	83:1	54	0.9	0.44	7.193	8
Vm2300	150:1	54	0.9	0.244	3.147	5
Vm2825	207:1	54	1.0	0.177	2.044	5
Avf1067a	150:1	57	0.9	0.244	2.016	4
Avf1200a	150:1	57	0.8	0.244	2.536	4
Avf1350a	68:1	54	0.9	0.537	7.692	8
Avf1700a	83:1	54	0.9	0.44	7.692	8
Avf1850a	83:1	54	1.0	0.44	6.933	8

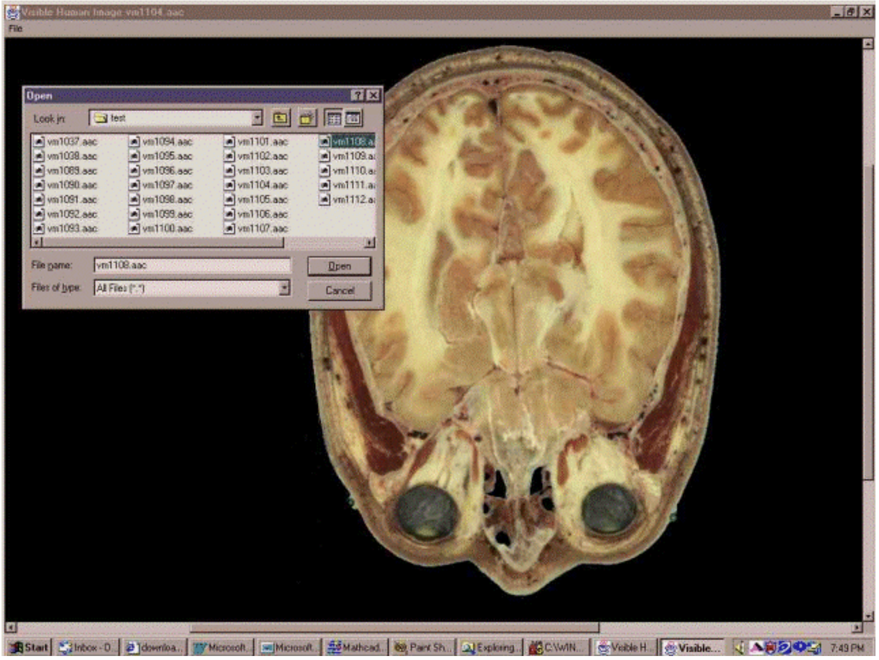
Figure 9. Overall analysis of the performance of the system.



You have Selected 400 Slices

Images a_vm1159 through a_vm1558

Figure 10. Snapshot of the selection/downloader applet.



Look in: test

vm1087.aac vm1088.aac vm1089.aac vm1090.aac vm1091.aac vm1092.aac vm1093.aac vm1094.aac vm1095.aac vm1096.aac vm1097.aac vm1098.aac vm1099.aac vm1100.aac vm1101.aac vm1102.aac vm1103.aac vm1104.aac vm1105.aac vm1106.aac vm1107.aac vm1108.a vm1109.a vm1110.a vm1111.a vm1112.a

File name: vm1108.aac

Files of type: All Files (*.*)

Figure 11. Snapshot of the viewer application.

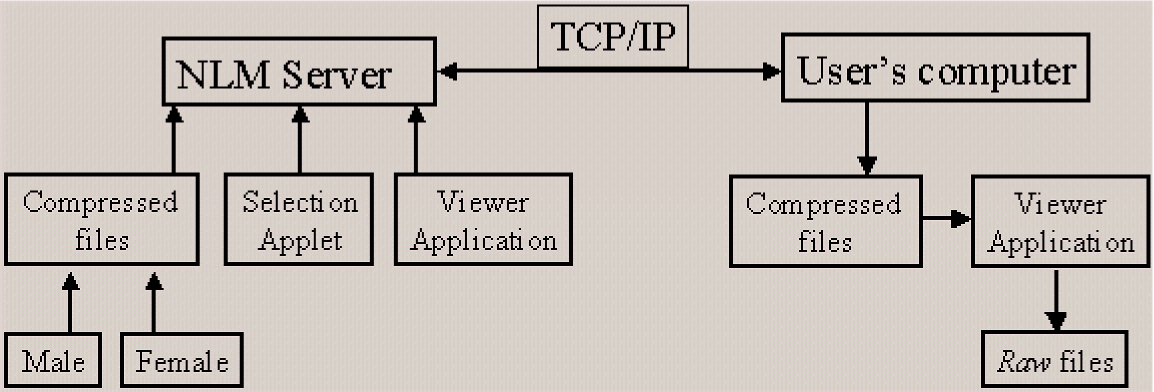


Figure 12. Block diagram of the system.

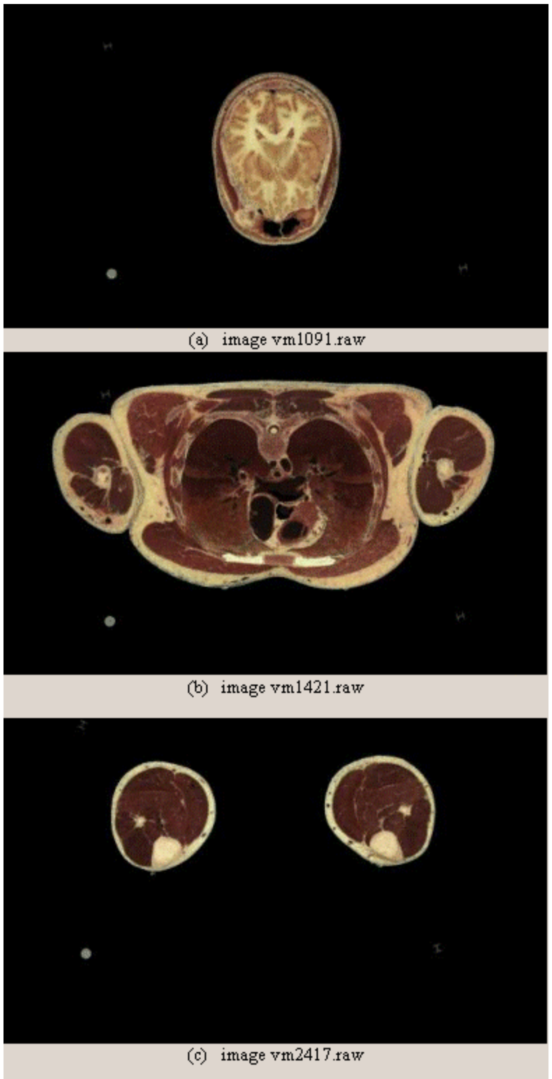
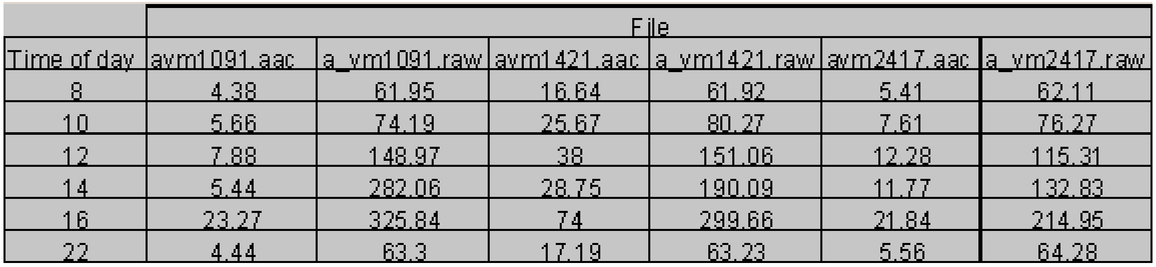


Figure 13. Male images used for testing from (a) head, (b) torso, and (c) knees.



	File					
Time of day	avm1091.aac	a_vm1091.raw	avm1421.aac	a_vm1421.raw	avm2417.aac	a_vm2417.raw
8	4.38	61.95	16.64	61.92	5.41	62.11
10	5.66	74.19	25.67	80.27	7.61	76.27
12	7.88	148.97	38	151.06	12.28	115.31
14	5.44	282.06	28.75	190.09	11.77	132.83
16	23.27	325.84	74	299.66	21.84	214.95
22	4.44	63.3	17.19	63.23	5.56	64.28

Figure 14. Transmission times.

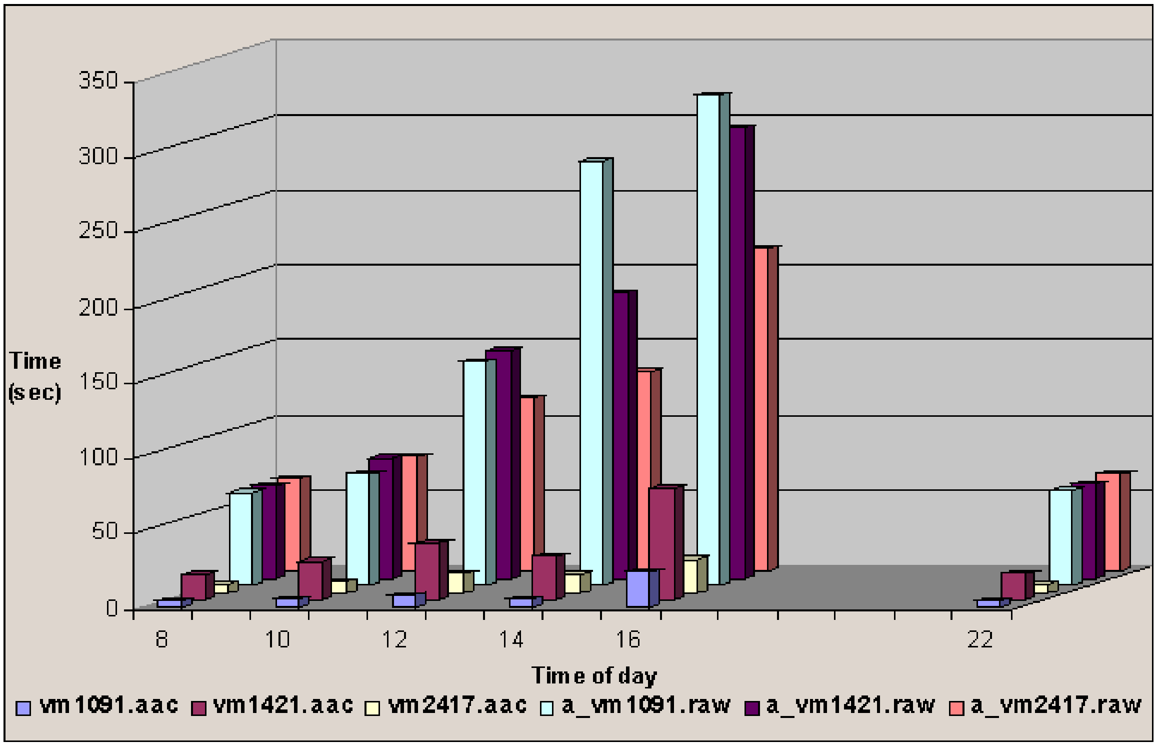


Figure 15. Plot of transmission times.